

(51) Int.Cl. ⁷	識別記号	F I	テマコード*(参考)
G 0 6 F 12/00	5 1 8	G 0 6 F 12/00	5 1 8 A
	5 3 3		5 3 3 J
	5 4 5		5 4 5 A
9/46	3 6 0	9/46	3 6 0 B
15/16	6 2 0	15/16	6 2 0 F
審査請求 有 請求項の数 5 O L (全 16 頁)			

(21) 出願番号 特願2000-353232(P2000-353232)
 (62) 分割の表示 特願平3-344685の分割
 (22) 出願日 平成3年12月26日(1991.12.26)

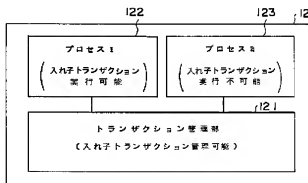
(71) 出願人 000003078
 株式会社東芝
 神奈川県川崎市幸区堀川町72番地
 (72) 発明者 金井 達徳
 神奈川県川崎市幸区小向東芝町1番地 株式会社東芝総合研究所内
 (72) 発明者 白木原 敏雄
 神奈川県川崎市幸区小向東芝町1番地 株式会社東芝総合研究所内
 (74) 代理人 100058479
 弁理士 鈴木 武彦 (外6名)

(54) 【発明の名称】 トランザクション処理の管理方法

(57) 【要約】

【課題】 入れ子トランザクション実行不可能なプロセスや入れ子トランザクション管理不可能なトランザクション管理部を持つ計算機が存在しても入れ子トランザクション処理を実行可能にするトランザクション管理方法を提供すること。

【解決手段】 入れ子トランザクション実行可能なプロセス1 2 2と入れ子トランザクション実行不可能なプロセス1 2 3の各々により、入れ子トランザクションのサブトランザクションを実行させ、入れ子トランザクション実行可能プロセス1 2 2により実行されたサブトランザクションのコミット処理を行い、入れ子トランザクションのトップレベルトランザクションを実行させ、実行されたトップレベルトランザクションのコミット処理を行い、トップレベルトランザクションのコミット処理完了と共に、入れ子トランザクション実行不可能プロセス1 2 3により実行されたサブトランザクションのコミット処理を完了させる。



【特許請求の範囲】

【請求項1】 入れ子トランザクション実行可能なプロセスと入れ子トランザクション実行不可能なプロセスとを含んだトランザクション処理システムにおけるトランザクション処理の管理方法であって、

前記入入れ子トランザクション実行可能なプロセスおよび前記入入れ子トランザクション実行不可能なプロセスの各々により、入れ子トランザクションのサブトランザクションを実行させ、

前記入入れ子トランザクション実行可能なプロセスにより実行されたサブトランザクションについてコミット処理を行い、

前記入入れ子トランザクションのトップレベルトランザクションを実行させ、

実行された前記トップレベルトランザクションのコミット処理を行い、

前記トップレベルトランザクションのコミット処理の完了と共に、前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理を完了させることを特徴とするトランザクション処理の管理方法。

【請求項2】 前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理を、前記トップレベルトランザクションのコミット処理と並行して行うことを特徴とする請求項10に記載のトランザクション処理の管理方法。

【請求項3】 前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の投票相を、前記入入れ子トランザクション実行可能なプロセスにより実行されたサブトランザクションについてのコミット処理と並行して行い、前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の決定相を、前記トップレベルトランザクションのコミット処理と並行して行うことを特徴とする請求項10に記載のトランザクション処理の管理方法。

【請求項4】 前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の投票相を、前記入入れ子トランザクション実行可能なプロセスにより実行されたサブトランザクションについてのコミット処理および前記入入れ子トランザクションのトップレベルトランザクションの実行と並行して行い、

前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の決定相を、前記トップレベルトランザクションのコミット処理と並行して行うことを特徴とする請求項10に記載のトランザクション処理の管理方法。

【請求項5】 前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについ

てのコミット処理の投票相を、前記入入れ子トランザクションのトップレベルトランザクションの実行と並行して行い、

前記入入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の決定相を、前記トップレベルトランザクションのコミット処理と並行して行うことを特徴とする請求項10に記載のトランザクション処理の管理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、分散処理環境に用いられる分散処理に適用されるトランザクション処理の管理方法に関する。

【0002】

【従来の技術】 分散処理環境においては、各種サービスを行なうサーバ・プロセスが複数の計算機上に分散して存在している。

【0003】 従って、ある計算機上のクライアント・プロセスが、他のサーバ・プロセスの提供する何らかのサービスを受けようとする場合には、そのサービスを提供しているサーバ・プロセスに対してサービス要求を伝えるようになる。このとき、受けたいサービスがどの計算機上のどのサーバ・プロセスで提供されているかを管理するため、従来では、ネーム・サーバ (Name Server) あるいはディレクトリ・サービス (Directory Service) と呼ばれるプロセスが用いられている。

【0004】 ネーム・サーバは、どのサーバ・プロセスがどのような名前前のサービスを提供しているかを管理しており、サービスを受けたいクライアント・プロセスがサービス名を伝えると、そのサービスを実行可能なサーバ・プロセスとこのサーバ・プロセスが存在する計算機とを答えるようになっていく。また、ネーム・サーバの中には各サービスに属性値を持たせ、サービスを受けたクライアント・プロセスの伝ええたサービス名と属性を基に、パターン・マッチングを行なうことで、能力を拡張したものも存在している。

【0005】 一方、このような分散処理には、システム信頼性を高めるためにトランザクション処理が用いられている。

【0006】 このようなトランザクション処理は、トランザクションと呼ぶ単位で処理を実行するようになっており、一つのトランザクションは、一つ以上のアプリケーション・プログラムやデータベース管理システムなどのプロセスを渡りあるいてこれらを実行することで、一連の処理を完了する。このとき一つのトランザクションを実行する複数のプロセスが、チャネルやネットワークを介して結合された複数の計算機に分散していても構わない。例えば、図25に示すように、トランザクション処理を行なう計算機34上には、処理を実行するプロセ

ス342、342…と、処理を管理するトランザクション管理部341が存在する。複数の計算機に分散したトランザクション処理を行なう場合には、図26に示すように、それぞれの計算機上にプロセスとトランザクション管理部351が存在し、各計算機上のトランザクション管理部351は通信し合いながらトランザクションの管理を行なう。トランザクション管理部は、各トランザクションが、以下の性質を持つようにその実行を制御するようにしている。

(1) 原子性 (atomicity) 処理が最後まで完全に実行されるか、もしくは完全な実行が不可能な場合には途中までの実行の影響が残らないことを保証する。

(2) 一貫性 (consistency) 処理することによって、一貫性のある状態から、一貫性のある状態へ遷移することを保証する。

(3) 分離性 (isolation) 並行して実行する複数の処理の結果は、それらを何らかの順序で直列に実行した場合と同じであることを保証する。

(4) 持続性 (durability) 完全に実行された処理の結果は、障害が発生しても復旧されることを保証する。

【0007】これらを実現するために、トランザクション管理部は各トランザクションの処理が終了した後に、コミット処理を行なう。コミット処理の方式は、2相コミットが代表的で良く使われる。他にも3相コミットをはじめとして様々な2相コミットを改良した方式が存在する。

【0008】例えば、図27のようにトランザクション管理部361と二つのプロセス362、363によるトランザクション処理の場合の2相コミット方式の動作を図29に示す。図29は時間の流れに沿った各構成要素間の相互作用を示している。ここではひとつのトランザクションがプロセス362とプロセス363の二つのプロセスによって処理されている。トランザクションの処理が終了すると、トランザクション管理部361はそのトランザクションの実行に関与したすべてのプロセス（ここではプロセス362とプロセス363）に対してコミット処理を行なう。コミット処理は投票相と決定相の2相からなる。まず投票相では各プロセスに対して、トランザクションをコミットすなわち正しく完了できるかどうかを問い合わせるVOTE-REQ要求を出す。各プロセスはこれに対してYESかNOを答える。YESと答える場合、各プロセスはトランザクションの実行結果をログに記録し、障害が発生しても処理結果を回復可能な状態にしてからYESと返答する。何らかの障害や異常があって正常に処理できない場合にはNOと返答する。すべてのプロセスからトランザクション管理部にYESあるいはNOという返答が返ってくると、投票相は終了して次の決定相に入る。すべてのプロセスがYESと答えた場合のみ、決定相ではトランザクション管理

部は各プロセスにCOMMIT要求を出す。COMMIT要求を受けたプロセスは、該当するトランザクションが正常に終了したものととして恒久的なデータの更新等の処理を行なう。

【0009】もし投票相で一つでもNOと返答したプロセスがあれば、トランザクションが正しく完了できないことになる。このような場合は、図30に示すようにアボート処理を行なう。すなわち、決定相ではすべてのプロセスに対してトランザクションの取り消しを指示するABORT要求を出す。ABORT要求を受けたプロセスは、該当トランザクションによるデータ等の更新を取り消す処理を行ない、トランザクションの原子性を保証する。

【0010】図27、図29、図30はプロセス362とプロセス363が同じ計算機上で動作している場合の例であった。図28のようにプロセス372とプロセス374が別の計算機で動作している場合は、計算機A上のトランザクション管理部371が図31のように計算機B上のトランザクション管理部373に処理要求を出し、トランザクション管理部373がトランザクション管理部371になり代わってプロセス374にコミット処理を行なう。

【0011】このようにして処理されるトランザクションは、図32のA、B…に示すように、互いに独立した処理の単位である。

【0012】一方、J. Eliot B. Moss 著の "Nested Transactions—An Approach to Reliable Distributed Computing" (The MIT Press, 1985) で開示された入れ子トランザクション (Nested Transaction) は、図33に示すように複数のトランザクション421～425を入れ子にして階層を持たせることによって、一つのトランザクションをより小さな単位のトランザクションの組合せとして実現する方式である。ここでは図34に示すように、階層の最上位のトランザクション431をトップレベル・トランザクションと呼び、下位のトランザクション432～434をサブ・トランザクションと呼ぶ。一つのトップレベル・トランザクションとその下位にあるすべてのサブ・トランザクションをまとめてトランザクション・ファミリーと呼ぶ。入れ子トランザクションにおいては、トランザクション・ファミリー間に入れ子でない非入れ子トランザクションと同じ扱いであるが、同じトランザクション・ファミリーに属するサブ・トランザクション間でも非入れ子トランザクションと同様に原子性や独立性を保証することにより、サブ・トランザクション間で並行実行したり、障害によるアボートをサブ・トランザクションのレベルで局所化することができるといった利点がある。

【0013】入れ子トランザクションを実行するために

は、図36に示すように入れ子トランザクションを実行可能なプロセス452と、入れ子トランザクションを実行不可能なプロセス453、入れ子トランザクションを管理可能なトランザクション管理部451が計算機上に存在しなければならない。プロセスが複数の計算機上に分散している場合には、図37に示すように、各計算機に入れ子トランザクションを実行可能なプロセス462とトランザクション管理部161、入れ子トランザクションを実行不可能なプロセス464とトランザクション管理部463を置く。

【0014】入れ子トランザクションのコミット処理は、非入れ子トランザクションの場合と異なり、トップレベル・トランザクションのコミット処理とサブ・トランザクションのコミット処理の二通りの場合に対応しなければならない。図38および図39に入れ子トランザクションのコミット処理の一例を示す。ここでは図35に示すようなトップレベル・トランザクションT441とそのサブ・トランザクションS442が、共にプロセスによって実行された場合を考えている。図38は二つのプロセスが同じ計算機上に存在して一つのトランザクション管理部がそれらを管理する場合である。図39は各プロセスが異なる計算機上に存在していて、それぞれの計算機上のトランザクション管理部によって管理される場合を示している。

【0015】まず、サブ・トランザクションのコミット処理を行なう。図38および図39の例ではトランザクション管理部がサブ・トランザクションがコミットしたことを内部に記録するだけで、プロセスに対しては何も指示しない。サブ・トランザクションのコミット時には、この例のように何もしない方式がよく用いられるが、通常の2相コミットと同様の処理をサブ・トランザクションに対して行うような実装法も可能である。

【0016】トップレベル・トランザクションのコミット処理も、非入れ子トランザクションの場合と同様である。ただし、トップレベル・トランザクションがコミットする場合には、トップレベル・トランザクションが更新したデータ等だけでなく、そのすべてのサブ・トランザクションが更新したデータ等も恒久的に記録されることを保証する必要がある。トップレベル・トランザクションがアボートする場合も同様に、そのすべてのサブ・トランザクションをアボートさせなくてはならない。このように、サブ・トランザクションに関してはコミットした状態は一時的なものでしかなく、より上位階層のトランザクションによってその状態が覆われる場合が存在する。入れ子トランザクションを実行可能なプロセスは、トランザクション管理部からの要求に答えてこのような処理を実行可能でなければならず、この点が入れ子トランザクションを実行不可能なプロセスとは異なる。

【0017】

【発明が解決しようとする課題】従来のトランザクシ

ン処理方式では、入れ子トランザクション処理を行なうためには、入れ子トランザクションを管理可能なトランザクション・マネージャと、入れ子トランザクションを実行可能なプロセスがなければならなかった。そのため、入ったトランザクションを実行不可能な従来のデータベース管理システムやアプリケーション・プログラムのプロセスを入れ子トランザクション処理に用いようとすると、入れ子トランザクションを実行できるようにプロセスのプログラムを書き換える必要があった。また、入れ子トランザクションを管理不可能な従来のトランザクション管理部とそれに管理される入れ子トランザクションを実行不可能なプロセスが存在する計算機と、入れ子トランザクションを管理可能なトランザクション管理部の存在する計算機とを結合したトランザクション処理は不可能であった。

【0018】本発明は、上記事情を考慮してなされたもので、入れ子トランザクションを実行不可能なプロセスや入れ子トランザクションを管理不可能なトランザクション管理部を持つ計算機が存在していても、入れ子トランザクション処理を実行可能にするトランザクションの管理方法を提供することを目的とする。

【0019】

【課題を解決するための手段】本発明は、入れ子トランザクション実行可能なプロセスと入れ子トランザクション実行不可能なプロセスとを含んだトランザクション処理システムにおける（入れ子トランザクションを管理可能なトランザクション管理手段による）トランザクション処理の管理方法であって、前記入れ子トランザクション実行可能なプロセスおよび前記入れ子トランザクション実行不可能なプロセスの各々により、入れ子トランザクションのサブトランザクションを実行させ、前記入れ子トランザクション実行可能なプロセスにより実行されたサブトランザクションについてコミット処理を行い、前記入れ子トランザクションのトップレベルトランザクションを実行させ、実行された前記トップレベルトランザクションのコミット処理を行い、前記トップレベルトランザクションのコミット処理の完了と共に、前記入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理を完了させることを特徴とする。

【0020】好ましくは、前記入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理を、前記トップレベルトランザクションのコミット処理と並行して行うようにしてもよい。

【0021】好ましくは、前記入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の投票相を、前記入れ子トランザクション実行可能なプロセスにより実行されたサブトランザクションについてのコミット処理と並行し

て行い、前記入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の決定相を、前記トップレベルトランザクションのコミット処理と並行して行うようにしてもよい。

【0022】好ましくは、前記入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の投票相を、前記入れ子トランザクション実行可能なプロセスにより実行されたサブトランザクションについてのコミット処理の決定相を、前記トップレベルトランザクションのコミット処理と並行して行うようにしてもよい。

【0023】好ましくは、前記入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の投票相を、前記入れ子トランザクションのトップレベルトランザクションの実行と並行して行い、前記入れ子トランザクション実行不可能なプロセスにより実行されたサブトランザクションについてのコミット処理の決定相を、前記トップレベルトランザクションのコミット処理と並行して行うようにしてもよい。

【0024】この結果、本発明のトランザクション処理の管理方法によれば、入れ子トランザクション処理を実行不可能なプロセスが入れ子になっていることを意識せずに実行したサブ・トランザクションについては、そのトランザクションのコミット処理の決定相の処理をトップレベル・トランザクションのコミット処理の決定相の処理時まで遅延させるため、そのトランザクションに関する原子性や持続性、一貫性、分離性といったトランザクションとしての性質を犠牲にすることがなくなる。また、入れ子トランザクションを実行不可能なプロセスの実行したトランザクションのコミット処理の投票相と、入れ子のトップレベル・トランザクションのコミット処理の投票相の投票結果を総合して双方のトランザクションの決定可能性を判断するため、入れ子トランザクション全体の原子性や持続性、一貫性、分離性といったトランザクションとしての性質も保証することができる。

【0025】

【発明の実施の形態】以下、本発明の一実施形態を図面に従い説明する。

【0026】上述した分散処理に適用されるトランザクション処理方式について説明する。

【0027】（第1の実施形態）図6は、トップレベル・トランザクションTとそのサブ・トランザクションSが、図3に示すようにトランザクション管理部121により入れ子トランザクションを実行可能なプロセス12

2と入れ子トランザクションを実行不可能なプロセス123を管理する場合のコミット処理を示している。

【0028】そして、図2は、この場合のトランザクション管理部121の構成を示している。この場合、トランザクション管理部121は、プロセス表1211とトランザクション表1212の二つの表を管理している。プロセス表1211には、その計算機上でトランザクション処理に関与するプロセスに対して入れ子トランザクションを実行可能か否かを記録している。トランザクション表1212は、現在処理中のトランザクションに対して、そのトランザクション識別子、階層関係で直接の親にあたるトランザクションのトランザクション識別子、そのトランザクションが実行中かコミット済みかを示す状態、そのトランザクションがどのプロセスによって処理されているか、そのトランザクションが他にどの計算機で処理されているかといった情報を記録している。そして、このようなトランザクション管理部121は、これら二つの表を参照しながら、後述するコミット処理を行なうようになる。

【0029】この場合、図5に示す流れに従って、トランザクション管理部121は、サブ・トランザクションSのコミット処理を行なう。この時はサブ・トランザクションSがコミットしたことをトランザクション管理部121がトランザクション表に記録するだけで、プロセス122やプロセス123に対して何も指示しない方式を採用している。この時点で従来の2相コミットと同等の処理を行なうような実施方式も可能である。その後、トップレベル・トランザクションTの処理が終了すればトップレベル・トランザクションのコミット処理を開始する。まず投票相では、Tの関与している入れ子トランザクションを実行可能なプロセス122に対してVOTE-REQ (T) を送る。さらに、Tのサブ・トランザクションSに関与した入れ子トランザクションを実行不可能なプロセス123にはVOTE-REQ (S) を送る（ステップS141～S143）。

【0030】そして、プロセス122からYES (T)、プロセス123からYES (S) という返答が返ってくれば次の決定相に移る。決定相では、プロセス122に対してCOMMIT (T) を送ると共に、トランザクションSに関与している入れ子トランザクションを実行不可能なプロセス123に対してCOMMIT (S) を送る。これによって、このトランザクション・ファミリー全体のコミット処理が完了する（ステップS144～S146）。また、トップ・レベルのコミット処理の投票相においてアボートが決定された場合には、入れ子トランザクションを実行可能なプロセスにトップレベル・トランザクションのアボート処理を行なうと共に、そのサブ・トランザクションを実行した入れ子トランザクションの実行不可能なプロセスに対して、該当するトランザクションのアボート処理も行なう（ステップS

147、S148)。

【0031】一方、図7は、図4に示すようにプロセス132とプロセス142が異なる計算機13、14上で実行され、それぞれの計算機13、14にトランザクション管理部131、141が存在する場合の入力になったトランザクションのコミット処理を示している。この場合、トランザクション管理部131、141は、図1(a)(b)に示すように上述したトランザクション管理部121と同様にプロセス表1311、1411とトランザクション表1312、1412のそれぞれ二つの表を管理している。

【0032】そして、トランザクション管理部131はそのトランザクション表1312を見て、トランザクションSが計算機14に拡がっていることを知り、計算機14上の入力子トランザクション処理を実行不可能なプロセス142に直接トランザクションSのコミット処理を行なうのではなく、トランザクション管理部141にトップレベル・トランザクションTのコミット処理を行なうように通信手段15によって伝える。それを受けたトランザクション管理部141が自身のトランザクション表1412とプロセス表1411を見て、入力子トランザクションを実行不可能なプロセス142に対してトランザクションSのコミット処理を行なうようになる。

【0033】なお、本実施形態のトランザクション管理部は、独立したプロセスとして実現することもできるし、ライブラリ等の形態でトランザクション処理を行なうプロセスにリンクするような実現方法も可能である。また、計算機のオペレーティング・システムにトランザクション管理部の機能を組み込むこともできる。また、本実施形態では、2相コミット方式に対して本発明を適用したものであるが、3相コミット等のように投票相と決定相を含むコミット処理方式に対しても同様に適用可能である。

【0034】(第2の実施形態) 上述の第1の実施形態においては、入力子トランザクションを実行不可能なプロセスで処理されたトランザクションのコミット処理の投票相と決定相は共にトップレベル・トランザクションのコミット処理時に行なった。

【0035】このようなサブ・トランザクションの投票相の処理は、トップレベル・トランザクションのコミット処理の決定相の処理が開始されるまでの任意の時点で行なわれてしまっても構わない。

【0036】そこで、図8に示すステップS171～S179の手順で処理を行なえば、例えば、図3に示すトランザクション管理部121が入力子トランザクションを実行可能なプロセス122と実行不可能なプロセス123を管理しているような場合は、図9に示すようにサブ・トランザクションのコミット処理時に入力子トランザクションを実行不可能なプロセスに対するコミット処理の投票相の処理を行ってしまうことができ、トップレ

ベル・トランザクションのコミット処理時に決定相の処理のみを行なうような実施方式を探ることができる。

【0037】また、図4に示す計算機13上には入力子トランザクションを実行可能なプロセス132が存在し、計算機14上には入力子トランザクションを実行不可能なプロセス142が存在し、双方の計算機13、14上にはトランザクション管理部131、141が存在するような場合は、図10に示すように、トランザクション管理部131はサブ・トランザクションSのコミット処理時に異なる計算機上のトランザクション管理部141に対してサブ・トランザクションSのコミット処理を依頼し、それを受けたトランザクション管理部141は自分が管理するプロセス142に対してサブ・トランザクションSのコミット処理の投票相を実行するようにできる。

【0038】(第3の実施形態) 上述の第2の実施形態においては、サブ・トランザクションSのコミット処理時にトランザクションSのコミット処理の投票相の処理を完了するのを待ち合わせたが、この処理はトップレベル・トランザクションのコミット処理の決定相の処理を開始するまでに完了していれば良い。

【0039】そこで、図11のステップS201～S208の手順にしたがって処理を行なえば、例えば図3のような場合には図12のようなコミット処理方式を行なうことができ、一方、図4のような複数の計算機にプロセスが分かれる場合は、図13のようにトランザクション管理部131はサブ・トランザクションSのコミット処理時にトランザクション管理部141にトランザクションSのコミット処理の投票相の開始を伝え、それを受けたトランザクション管理部141が自分の管理する入力子トランザクションを実行不可能なプロセス143にコミット処理の投票相の開始を伝えるようにすればよい。

【0040】(第4の実施形態) 上述の第2の実施形態においては、サブ・トランザクションSのコミット処理時にトランザクションSに対するコミット処理の投票相を起動したが、この投票相の起動は、図14に示すように、サブ・トランザクションのコミット処理が終了してからでも構わない。これは、図4のようにトランザクションが複数の計算機上で処理されている場合でも同様であり、図15のようなコミット処理方式を実施することができる。

【0041】例えば、図15のように複数の計算機にトランザクションが拡がっている場合には、サブ・トランザクションのコミット処理が終了した後に計算機13から計算機14に対して何らかのメッセージを送る時に便乗してサブ・トランザクションSのコミット処理の投票相の開始要求をトランザクション管理部131からトランザクション管理部142に送ることができる。

【0042】(第5の実施形態) この場合、図16は、

11

入れ子トランザクションを管理可能なトランザクション管理部251とプロセス252を有する計算機25と、入れ子トランザクションを管理不可能な従来のトランザクション管理部261とプロセス262を有する計算機26とからなるトランザクション処理装置を示している。そして、トランザクション管理部251、261の構成を図17(a)(b)、この時のコミット処理の手順を図18に、コミット処理方式を図19にそれぞれ示している。

【0043】この場合、サブ・トランザクションSのコミット処理時にはトランザクション管理部251はそのトランザクションがコミットしたことを記録するだけで他は何もしない。このときトランザクション管理部261は、サブ・トランザクションSはまだ実行中の状態として管理している。トップレベル・トランザクションTのコミット処理時には、トランザクション管理部251はまず入れ子トランザクションを実行可能なプロセス252に対してVOTE-REQ(T)を送る。次にノード表を見て、入れ子トランザクションを管理不可能なトランザクション管理部の存在する計算機にサブ・トランザクションSが広がっていることを知り、トランザクション管理部261にVOTE-REQ(S)を送る(図18のステップS271～S273)。

【0044】トランザクション管理部261はそのトランザクション表を見て、トランザクションSがプロセス262で実行されていることを知り、プロセス262にVOTE-REQ(S)を送る。プロセス262が投票相の処理を完了してYES(S)をトランザクション管理部2に返してくると、トランザクション管理部261はトランザクション管理部251にYES(S)を返す。トランザクション管理部251は、プロセス252からのYES(T)とトランザクション管理部261からのYES(S)を受け取ると、トップレベル・トランザクションの決定相の処理を行なう。決定相では、トランザクション管理部251はプロセス252にCOMMIT(T)を送る。次にトランザクション管理部261にはCOMMIT(S)を送る。それを受けたトランザクション管理部261はプロセス262に対してCOMMIT(S)を送り、決定相の処理を完了する(図18のステップS274～S276)。また、トランザクション管理部251は、プロセス252からのYES(T)とトランザクション管理部261からのYES(S)を受け取らない場合は、該当するトランザクションのアボート処理も行なう(ステップS277、S278)。

【0045】(第6の実施形態) 上述の第5の実施形態においては、計算機26に拡がったトランザクションSのコミット処理をすべてトップレベル・トランザクションTのコミット処理時に行なう。このうち、投票相の処理は、トップレベル・トランザクションのコミット処

12

理の決定相の処理を開始するまでに終えていなければならない。

【0046】そこで、図20のステップS291～S299に示すコミット処理の手順、図21に示すコミット処理方式をそれぞれ用いれば、サブ・トランザクションのコミット処理時にトランザクションSの投票相の処理を行なうことができる。

【0047】(第7の実施形態) 上述の第5の実施形態においては、計算機26に拡がったトランザクションSのコミット処理をすべてトップレベル・トランザクションTのコミット処理時に行なう。このうち、投票相の処理は、トップレベル・トランザクションのコミット処理の決定相の処理を開始するまでに終えていなければならない。そこで、図22のステップS311～S318に示すコミット処理の手順、図23に示すコミット処理方式を用いても、サブ・トランザクションのコミット処理時にトランザクションSの投票相の処理の開始を指示することが可能になる。

【0048】(第8の実施形態) 上述の第5の実施形態においては、計算機26に拡がったトランザクションSのコミット処理をすべてトップレベル・トランザクションTのコミット処理時に行なう。このうち、投票相の処理は、トップレベル・トランザクションのコミット処理の決定相の処理を開始するまでに終えていなければならない。そこで、図24に示すコミット処理方式を用いれば、サブ・トランザクションのコミット処理以降の任意の時点でトランザクションSの投票相の処理の開始を指示することが可能になる。この処理の開始指示のタイミングは、計算機25から計算機26へ何らかの通信が発生した時点で便乗して知らせれば良い。

【0049】その他、本発明は上記実施形態にのみ限定されず、要旨を変更しない範囲で適宜変形して実施できる。

【0050】

【発明の効果】以上述べたように、本発明のトランザクション処理の管理方法によれば、入れ子トランザクションを実行可能なプロセスと実行不可能なプロセスが混在する場合でも、入れ子トランザクション処理が可能になり、また、入れ子トランザクションが実行不可能なプロセスは非入れ子トランザクション処理に用いていたものをそのまま使用することも可能になる。

【図面の簡単な説明】

【図1】本発明の第1の実施形態のトランザクション管理部の構成を示す図

【図2】本発明の第1の実施形態のトランザクション管理部の構成を示す図

【図3】本発明の第1の実施形態の入れ子トランザクション処理装置を示す構成図

【図4】本発明の第1の実施形態の分散入れ子トランザクション処理装置を示す構成図

【図5】本発明の第1の実施形態の動作を説明するためのフローチャート

【図6】本発明の第1の実施形態でのコミット処理を説明するための図

【図7】本発明の第1の実施形態でのコミット処理を説明するための図

【図8】本発明の第2の実施形態の動作を説明するためのフローチャート

【図9】本発明の第2の実施形態でのコミット処理を説明するための図

【図10】本発明の第2の実施形態でのコミット処理を説明するための図

【図11】本発明の第3の実施形態の動作を説明するためのフローチャート

【図12】本発明の第3の実施形態でのコミット処理を説明するための図

【図13】本発明の第3の実施形態でのコミット処理を説明するための図

【図14】本発明の第4の実施形態でのコミット処理を説明するための図

【図15】本発明の第4の実施形態でのコミット処理を説明するための図

【図16】本発明の第5の実施形態の分散入れ子トランザクション処理装置を示す構成図

【図17】本発明の第5の実施形態のトランザクション管理部の構成を示す図

【図18】本発明の第5の実施形態の動作を説明するためのフローチャート

【図19】本発明の第5の実施形態でのコミット処理を説明するための図

【図20】本発明の第6の実施形態の動作を説明するためのフローチャート

【図21】本発明の第6の実施形態でのコミット処理を説明するための図

【図22】本発明の第7の実施形態の動作を説明するためのフローチャート

【図23】本発明の第7の実施形態でのコミット処理を説明するための図

【図24】本発明の第8の実施形態でのコミット処理を

説明するための図

【図25】従来のトランザクション処理装置を示す構成図

【図26】従来の分散トランザクション処理装置を示す構成図

【図27】従来の他のトランザクション処理装置を示す構成図

【図28】従来の他の分散トランザクション処理装置を示す構成図

10 【図29】従来のトランザクション処理装置のコミット処理を説明するための図

【図30】従来のトランザクション処理装置のアポード処理を説明するための図

【図31】従来のトランザクション処理装置のコミット処理を説明するための図

【図32】独立して処理されるトランザクションの状態を示す図

【図33】トランザクションを入れ子にして階層を持たせた構成を示す図

20 【図34】トランザクションを入れ子にして階層を持たせた構成を示す図

【図35】入れ子トランザクションを示す図

【図36】入れ子トランザクション処理装置を示す構成図

【図37】分散入れ子トランザクション処理装置を示す構成図

【図38】入れ子トランザクションのコミット処理の一例を示す図

30 【図39】入れ子トランザクションのコミット処理の一例を示す図

【符号の説明】

121…トランザクション管理部

1211…プロセス表

1212…トランザクション表

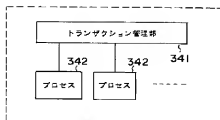
122、123…プロセス

13、14、25、26…計算機

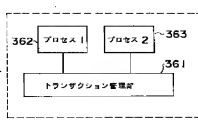
131、141、251、261…トランザクション管理部

132、142、252、262…プロセス

【図25】



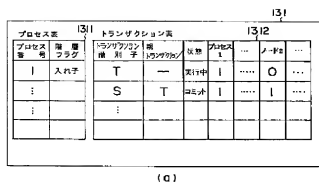
【図27】



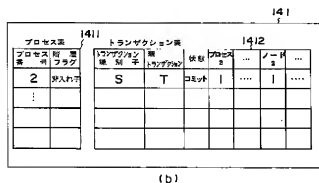
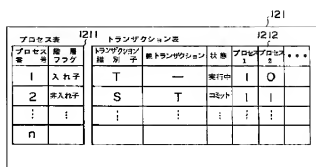
【図35】



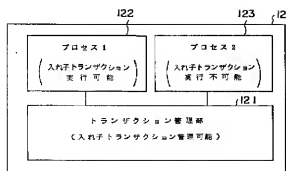
【図1】



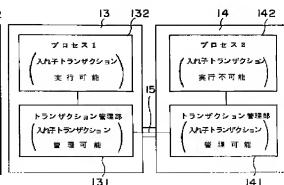
【図2】



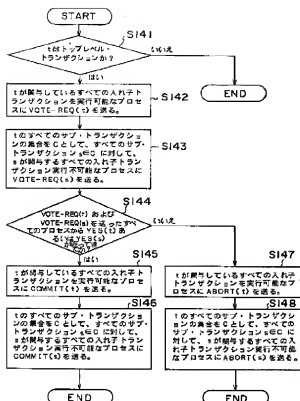
【図3】



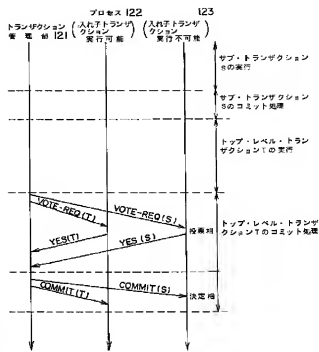
【図4】



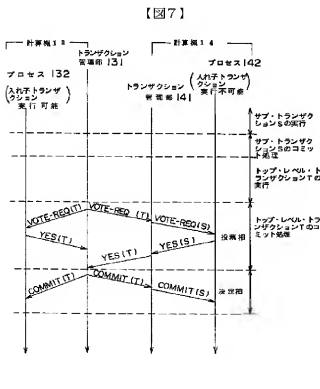
【図5】



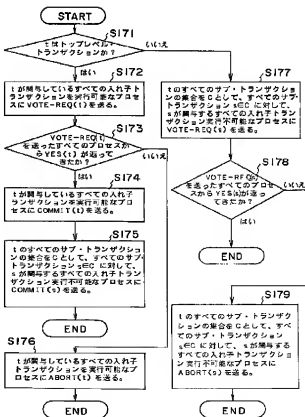
【図6】



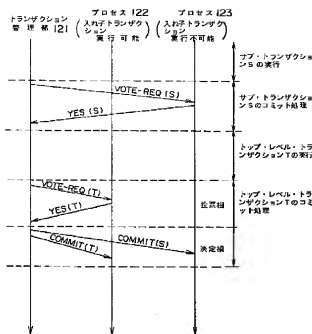
【図8】



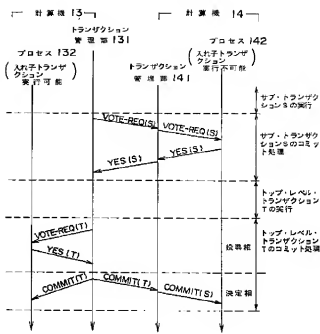
【図7】



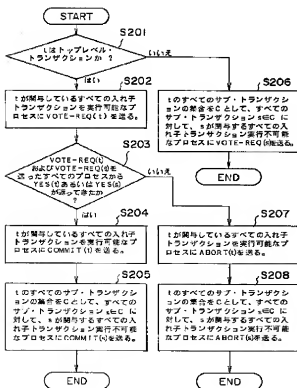
【図9】



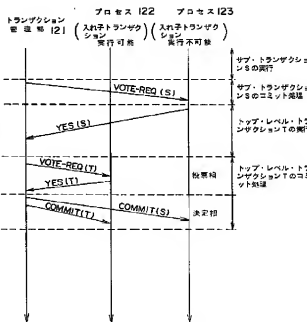
【図10】



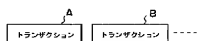
【図11】



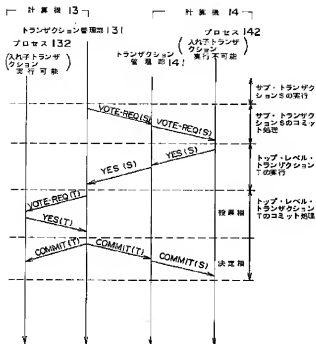
【図12】



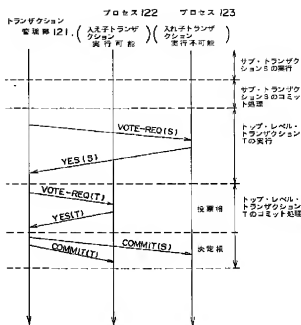
【図13】



【図13】

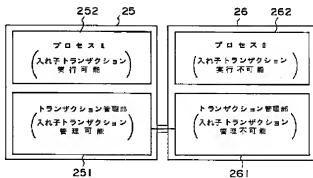
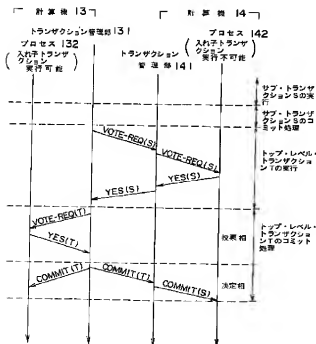


【図14】

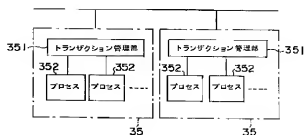


【図16】

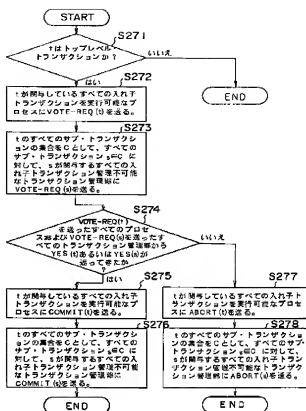
【図15】



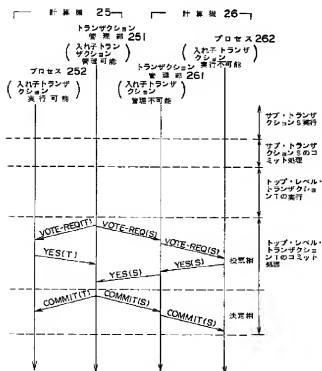
【図26】



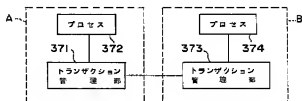
【图18】



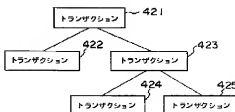
【图19】



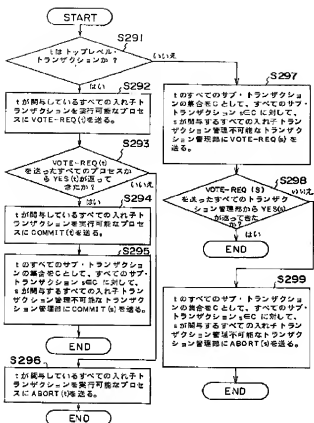
【图28】



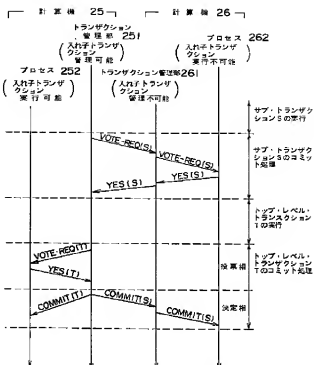
【图3-3】



【图20】

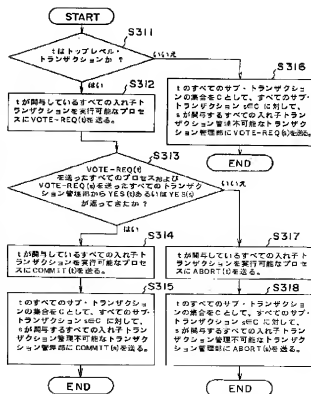
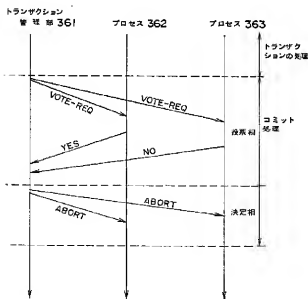


【图21】

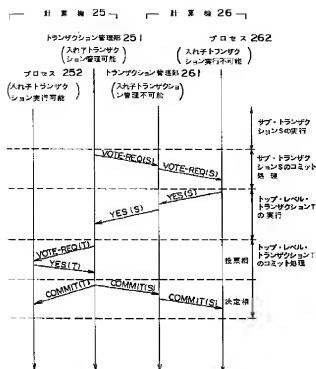


【图22】

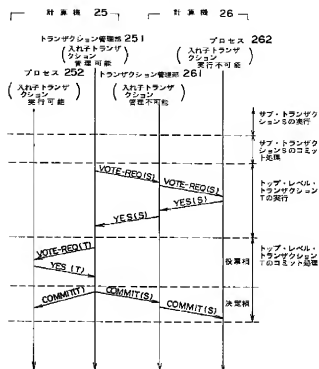
【图30】



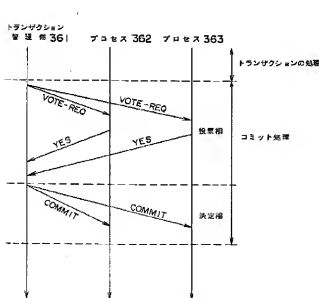
【図23】



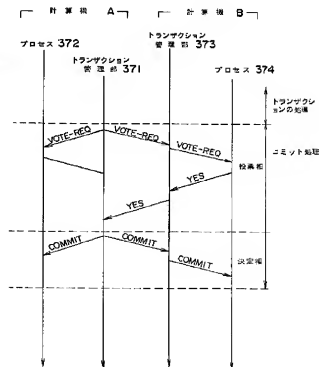
【図24】



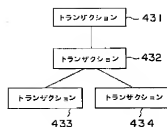
【図29】



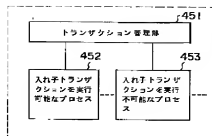
【図31】



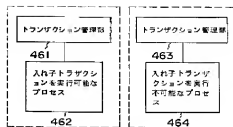
【図34】



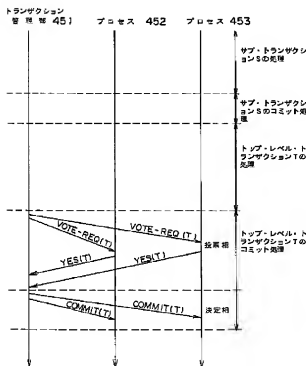
【図36】



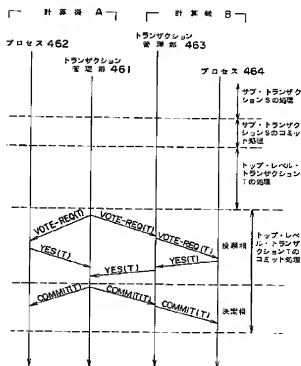
【図37】



【図38】



【図39】



PAT-NO: JP02001188696A
DOCUMENT-IDENTIFIER: JP 2001188696 A
TITLE: MANAGING METHOD OF
TRANSACTION PROCESSING
PUBN-DATE: July 10, 2001

INVENTOR-INFORMATION:

NAME	COUNTRY
KANAI, TATSUNORI	N/A
SHIROKIBARA, TOSHIO	N/A

ASSIGNEE-INFORMATION:

NAME	COUNTRY
TOSHIBA CORP	N/A

APPL-NO: JP2000353232
APPL-DATE: December 26, 1991

INT-CL (IPC): G06F012/00 , G06F009/46 ,
G06F015/16

ABSTRACT:

PROBLEM TO BE SOLVED: To provide a transaction managing method to execute a nested transaction processing even when a computer with a process incapable of executing the nested transaction processing and a transaction managing part incapable of managing transaction processing exists.

SOLUTION: Sub-transaction of the nested transaction is executed by each of a process 122 capable of executing a nested transaction and a process 123 incapable of executing the nested transaction, a commitment processing of the sub-transaction executed by the process 122 capable of executing the nested transaction is performed, top level transaction of the nested transaction is executed, the commitment processing of the executed top level transaction is performed, the commitment processing of the sub-transaction executed by the process 123 incapable of executing the nested transaction is completed simultaneously with completion of the commitment processing of the top level transaction.

COPYRIGHT: (C)2001,JPO